DTIC FILE COPY

(4)

AD-A200 724

Georgia Institute of Technology

LOCATION ISSUES IN GUARANTEED TIME DISTRIBUTION SYSTEMS

by
Ananth V. Iyer
H. Donald Ratliff

PDRC 87-08

PRODUCTION and DISTRIBUTION RESEARCH CENTER

SCHOOL OF INDUSTRIAL
AND SYSTEMS
ENGINEERING
GEORGIA INSTITUTE OF TECHNOLOGY
A UNIT OF THE UNIVERSITY SYSTEM
OF GEORGIA
ATLANTA, GEORGIA 30332

PDRC Report Series 87-08
August 1987

LOCATION ISSUES IN GUARANTEED TIME DISTRIBUTION SYSTEMS

by

Ananth V. Iyer
H. Donald Ratliff

PDRC 87-08

School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, GA 30332

This work was supported by the Office of Naval Research under Contract No. N00014-86-K-0173. Reproduction is permitted in whole or part for any purpose of the U.S. Government.

LOCATION ISSUES IN GUARANTEED TIME DISTRIBUTION SYSTEMS

Ananth V. Iyer† H. Donald Ratliff‡

Abstract

The guaranteed time distribution problem consists of organizing the distribution system so that movement between any source destination pair (serviced by the system) can be completed within some guaranteed time (T). Various express mail services (e.g. Federal Express) are examples of such distribution systems. The movement of items between source and destination consists of a combination of different modes of travel with varying speeds (i.e. trucks, airplanes etc.). Decisions have to be made regarding location accumulation points (called local centers) at which the different modes of travel are affected that provide the best (minimum) time guarantee for travel between any source-destination pair. Typical distribution systems tend to have a tree structure; for such systems we provide polynomial time optimal algorithms which locate these accumulation points and provide minimum time guarantee. We also address related models with varying degrees of coordination and present optimal algorithms for their solution.

†Graduate School of Business, The University of Chicago, 1101 East 58th Street, Chicago, IL 60637
‡School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30332
This work was supported by the office of Naval Research under Contract No. N00014-86-K-0173.

LOCATION ISSUES IN GUARANTEED TIME DISTRIBUTION SYSTEMS

1.0 Introduction

The guaranteed time distribution problem consists of organizing the distribution system so that movement between any source destination pair (serviced by the system) can be completed within some guaranteed time (T). Various express mail services (e.g. Federal Express) are examples of such distribution systems.

Movement of items between source and destination consists of travel (e g by truck) to an accumulation point (called a **local center**) where there is a change of mode to a rapid transit mode (e.g. airplanes) which takes the item to its destination local center. At the destination local center there is again a change of mode (e.g. to trucks) to move the items to their destination. Decisions in such systems concern location of a given number of local centers to provide the best (minimum) time guarantee.

We provide polynomial time algorithms that locate a given number of local centers and provide the best system time guarantee when the routes linking the points (served by the system) form a tree structure. Alternatively, given a desired system time guarantee (T), our algorithms would locate the minimum number of local centers required to affect the distribution.

1.1 Problem Definition

Consider a distribution system that services a set of n customers. Requests for service originate at each of these customer locations. A service request specifies a flow from an origin to a destination. There is a potential flow between all pairs of customers. We wish to satisfy these service requests through the use of k local centers. Local centers correspond to accumulation points in the system and contain sorting equipment etc. required to affect the distribution.

The domain of a local center refers to all points allocated to that center. All flows originating from and destined for points in the domain of a local center must pass through the local center. Flow between any pair of points i and j consists of movement to a local center S_i, transit from S_i to center S_j (in whose domain j occurs) and movement from S_j to the destination j.

There is a change of mode at local centers that enables rapid transit between these centers. We model this change of mode through a **"speed up" factor = a** (<= 1). The parameter "a" is a multiplier on the distances between local centers that efffectively reduces the distance between local centers.

Thus the travel time between points i and j is expressed as

$$T(i,j) = d(i,S_i) + a * d(S_i,S_j) + d(j,S_j)$$

The objective considered here is to "locate" the centers and "allocate" customers to these centers so as to minimize the maximum travel time (T(i,j)) between any pair of customer locations. This objective is precisely the required minimum system time guarantee (T).

1.2 Example Problem

Figure 1 shows an example graph whose nodes correspond to customer locations. Every arc (i,j) in the graph has a weight d(i,j) that is the time to go from i to j (or j to i). Paths in the graph represent feasible travel routes for flows.

In this case, S_1 and S_2 are the two local centers. The domain of each of the centers is indicated by dashed lines. Travel between points 1 and 2, which are in the domain of center S_1, consists of travel from 1 to S_1 and

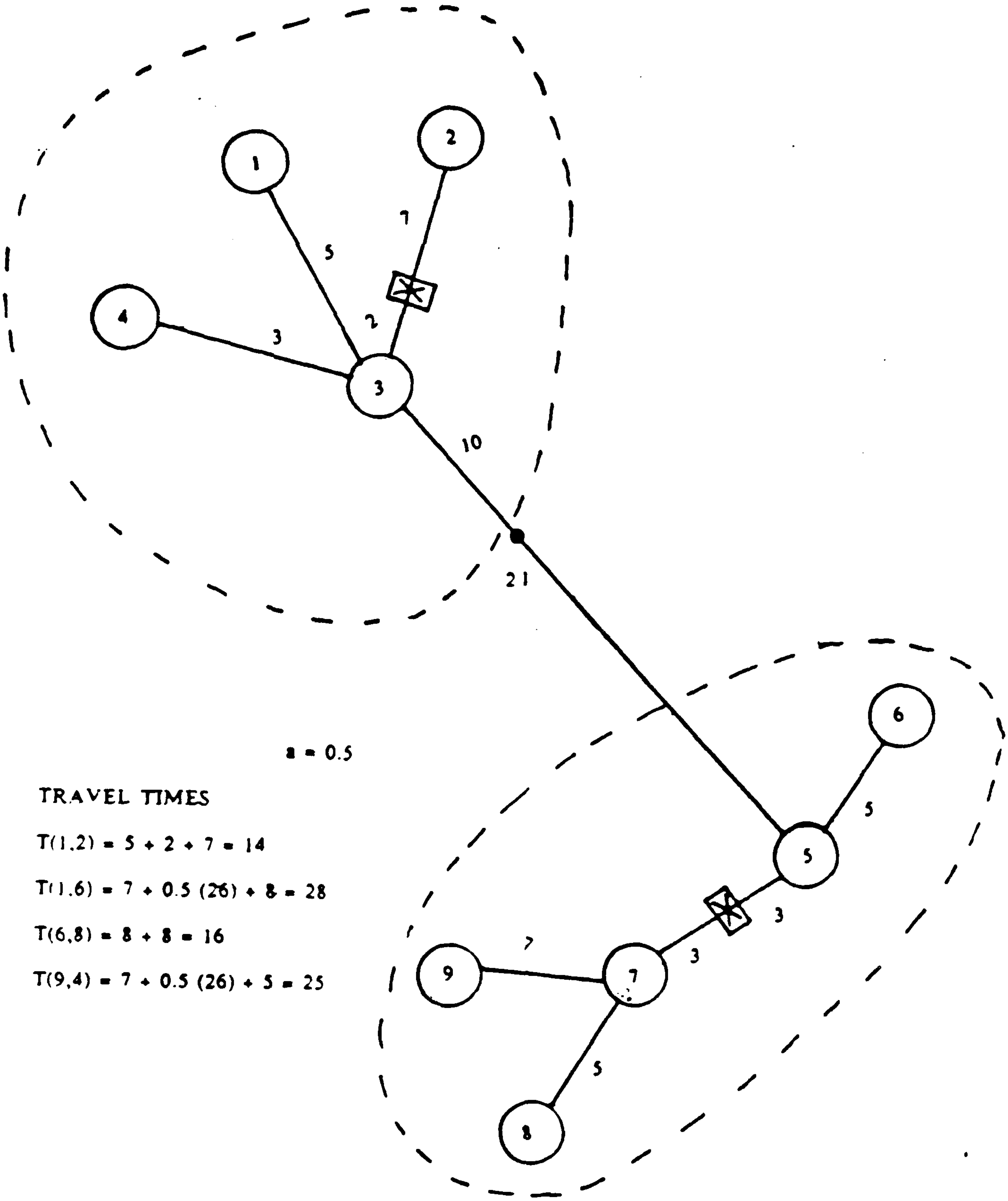


Figure 1 An example problem and center locations

then from S_1 to 2. Therefore travel time between points 1 and 2 $= d(1,S_1) + d(2,S_1)$. Travel between points 1 (in the domain of S_1) and 6 (in the domain of S_2) consists of travel from 1 to S_1, travel from S_1 to S_2 and finally travel from S_2 to point 6. Rapid transit between local centers implies a "speed up" in travel time by a factor of a ($a <= 1$). Therefore travel time between points 1 and 6 $= d(1,S_1) + a\, d(S_1,S_2) + d(6,S_2)$.

The maximum travel time is between points 1 and 6 with a value of 28. Thus the system time guarantee for the configuration in Figure 1 is $T = 28$ units.

1.3 Local Centers

A local center consolidates flows from points within its domain to the rest of the system. It serves as a redistribution point for flows originating from points in its domain. It also receives flows from the rest of the system, destined for points in its domain, and distributes the flows to points in its domain. The local centers are points at which there is a change of mode. This implies that consolidated flows from a local center to the rest of the system use faster mode of travel such as express trucks, airplanes etc. Sorting equipment required to sort flows for distribution need only be located at local centers. Thus we reduce the amount of sorting equipment in the distribution system. Finally, by consolidating flows at the local centers before using a faster mode, we could take advantage of economies of scale in the transportation between centers.

Our models specifically address improvements in time guarantee resulting from use of rapid transit between local centers. We model the tradeoffs between the number of sorting equipment sets available and the corresponding time guarantee by providing the minimum time guarantee for a given number k of local centers. We partially address the economies of scale issues through the use of a tree structured transportation configuration.

1.4 Tree structured transportation configuration

We consider the problem when customer locations are linked by a tree structure. This means that there is a unique simple path between any two customer locations and all customers are connected by the transportation system. Figure 1 is an example tree structured configuration linking the nine customer locations.

Tree structured configurations are common in distribution systems. Federal Express uses a single hub located at Memphis to perform all sortation for re-distribution. Flows into this sortation center, from the rest of the country, follow unique paths and thus generate a tree structured transportation configuration (by definition a connected set of points with unique paths between them forms a tree graph).

Many distribution systems use a "hub and spoke" type of route layout which also happens to be a tree graph.

Furthermore, from an algorithmic perspective, we can show that the center location problem is NP-Complete when the transportation configuration is a general network. However, we provide polynomial time algorithms for the tree structured transportation systems.

1.5 Designing distribution systems

There remains, however, the issue of defining the tree structured transportation configuration for a given set of customer locations. In this paper we assume that the transportation configuration is provided as input to the problem. This implies that for a given transportation configuration (or route structure) our algorithms would generate the best location decisions. The algorithms thus serve to evaluate alternative transportation configurations and could therefore be used in the design of such distribution systems.

1.6 Literature Survey

O'Kelly '86 discussed the location of two interacting hub facilities under a Euclidean metric. The problem was solved by considering all possible non overlapping partitions of customers. Goldman '69 and Hakimi and Maheshwari '72 discuss a k median location problem with interaction on a general graph, showing that it is sufficient to consider median locations at nodes of the graph. However, no algorithm was provided to solve the problem even for tree structures.

Extensive literature is available for the k median and k center problem without interaction. Tansel, Francis and Lowe '83 and Kariv and Hakimi '79 provide such bibliographies.

Other related application areas are considered by Coffman et al '85, Choi and Hakimi '87 and Iyer,Ratliff and Vijayan'87.

1.7 Problem on a general network

The above problem defined on a general network would restrict travel to be along edges of the network and locations to be on nodes or edges of the network. However this problem is NP-Complete for the co-ordination strategies that we consider. This is because setting $a = 0$ reduces our problem to the k-center problem on a general network which is known to be NP-Complete (Kariv and Hakimi '79).

We examine tree structured location problems because there are known polynomial algorithms for non-interacting center and median problems (Kariv and Hakimi '79) which indicate a potential for polynomial algorithms for the interacting location problems on tree.

2.0 Centralized sortation system (CSS)

2.1 Description of the CSS operation

We illustrate the operation of a CSS in Figure 2. This system serves 11 points, has 3 local centers S_1, S_2 and S_3 and a central hub located at C. We also assume that the speed up factor a = 0.2. Flow between points 1 and 4 is affected as as follows. At time 0, items flow from 1 to the local center S_1. At S_1 we accumulate all items flowing from points in its domain (i.e. points 1,2,3) to the rest of the system (i.e points 4 to 11). These accumulated items are sent (all flows in S_1 get accumulated at time = 10 units) by rapid transit to the central hub C reaching it at time = 10 + 0.2 * 25 = 15 units.

At central hub C, the items interchanged across local center domains are accumulated. Accumulation at the central hub is completed at time = 18 units (i.e. when the items from 10 reach C) and regrouped by destination local center domains.

These regrouped sets of items are again sent by rapid transit to their destination local centers, thus items for 4 along with all flows from the rest of the network (i.e. 1 to 3 and 8 to 11) destined for the domain of S_2 is sent to local center S_2 and reaches it at time = 18 + 0.2 * 18 = 21.6 time units. From S_2 the items are sent to their destinations i.e. flow (1,4) reaches its destination node 4 at time = 27.6 units.

However, the last item to reach its destination is again that destined for node 10 at time = 18 + 0.2 * 15 + 15 = 36 units. Thus the system time guarantee = 36 units.

Thus for the CSS guaranteed time for a given location of local centers and central hub is

$2 * \text{Max}_{(\text{all } i)} \{ d(i,S_i) + a * d(S_i,C) \}$

In a CSS system, the guaranteed time to distribute flow between any pair of points = 2 * Guaranteed time to arrive at the central hub from any customer location. We can thus consider the problem of locating k centers and allocating customers to centers so as to minimize the maximum time to get to the central hub from any customer location as it would correspondingly minimize the overall distribution objective.

Typical guaranteed time delivery systems in the postal industry such as Federal Express (with a central hub in Memphis) utilize such a central hub type of system.

2.2 CSS resource requirements

The CSS has only one main sortation center located at the central hub because the local centers have only to distinguish between flows destined for different destinations in their domain.

It also implies that we have k rapid transportation resource units (i.e. k airplanes) which provide rapid transit for flows (in parallel) from a local center to the central hub and the destination flows from the central hub back to the local centers.

2.3 Analysis of the model

2.3.0 Solution strategy

Consider the CSS location problem with the centers restricted to lie on nodes of the tree graph. Suppose we have to locate k centers and allocate customers to centers with time guarantee <= T1. Thus we require a location of centers and an allocation of customers so that the time to get to the central hub from any of the customers is <= T = T1/2.

Given the central hub C, only nodes <= $\{T - a * d(C,i)\}$ from i could be allocated to a center located at node i, where d(C,i) is the distance from node i to center location C (unique for tree graphs). Also, any pair of nodes in the domain of a local center should be <= T1 (= 2 * T) units apart.

We ensure these constraints are satisfied by specifying the domain of a center located at node i as the set of customers <= T - a d(C,i) units of travel from node i.

We solve the CSS problem (with central hub specified) as follows

(i) For a given value of T (time guarantee/2) we provide the minimum number of local centers required to ensure flows reaching the hub in <= T.

(ii) We perform a logarithmic search over T to determine the minimum time T that could be generated with k centers.

2.3.1 <u>Balanced matrix structure</u>

Given a value of T (i.e time to accumulate flows at the central hub) and location of the central hub C, the minimum number of centers required can be expressed as a covering problem.

The covering problem identifies the minimum number of centers required to cover every customer j. A customer j is covered if there is at least one center i at a distance of <= T - a d(C,i) from j. The covering problem can thus be expressed as follows:

Let $x_j = 1$ if j is a local center, 0 otherwise

N_j = set of those nodes i such that d(i,j) <= T - a * d(j,C)

Problem C1: $\quad \text{Min} \sum_j x_j$

$$\text{s.t. } A\,x >= 1 \quad \text{for all } i = 1,2,..n$$

$$x_j >= 0\ ;\ xj : 0/1 \text{ integer for } j = 1,2,..n$$

A : n x n matrix with entries of 0 or 1;

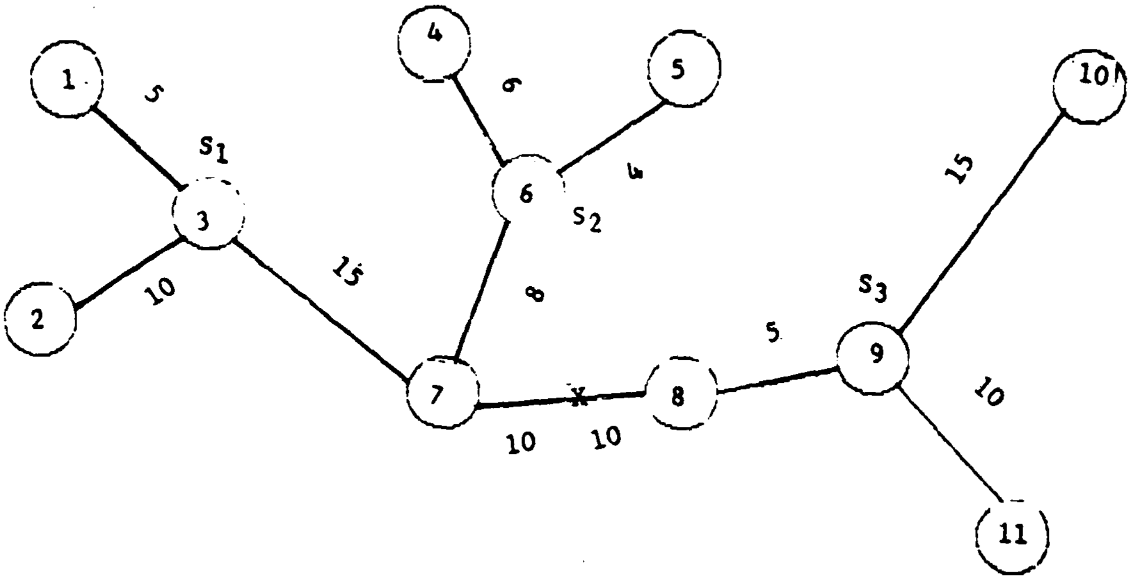


Figure 2: Example CSS with hub located at X i.e. on (7,8) 10 units from 7.

	1	2	3	4	5	6	7	8	9	10	11
1	1		1								
2		1									
3	1	1	1				1				
4				1	1	1	1				
5				1	1	1	1				
6				1	1	1	1				
7					1	1	1				
8								1	1		
9								1	1		1
10									1	1	
11								1	1		1

Figure 3: The A matrix in C1 corresponding to T = 18 (T1 = 36) for the data in Figure 2

$A[i,j] = 1$ if i in N_j; $A[i,j] = 0$ otherwise.

Figure 3 shows the A matrix for the CSS problem in Figure 2 with $T = 18$.

Solution of the covering problem C1 yields the minimum number of local centers required to provide a time guarantee of T1 and thus provides the solution to step (i) of the solution strategy in 2.3.0

The matrix (A) in this covering problem is called a node-star adjacency matrix because the columns (say column j) in this matrix form a star type structure i.e. include all those nodes within a distance D_j ($= T - a * d(j,C)$) from node j.

Kolen '82 has shown that such node star adjacency matrices are totally balanced. Totally balanced matrix structure implies that the problem feasible region has all integer extreme points and hence can be solved by linear programming.

Kolen'82 and Broin and Lowe'86 provide algorithms to solve the covering problems in $O(n^2)$ time. These algorithms permit inclusion of costs for locating local centers at each vertex and thus generate the minimum cost covering of all customer locations for a given time guarantee.

2.3.2 <u>Algorithm CSS</u>

Therefore an algorithm to solve the CSS location model is as follows

(1) Set a value for the objective function T = longest path/2 = $L/2$.

(2) For a given T, set up the covering problem C1 and solve using either one of the algorithms mentioned above (or linear programming). We thus generate the minimum number of centers required to generate a time guarantee of T.

(3) Do a binary search over T to determine the minimum time guarantee that can be generated with k centers for a given hub location. The overall time

guarantee = 2 * T.

Running time of the algorithm (using the algorithms for solving the totally balanced matrices) = O (n^2 log (L)).

2.4 CSS time guarantee versus minimum number of centers

As the time guarantee (T) for a CSS system decreases, we expect the number of local centers required to generate that time guarantee to increase. Figure 4 shows the minimum number of local centers (Y-axis) required to generate the time guarantee in the X-axis for the problem in Figure 2.

3.0 **Decentralized Sorting System (DSS)**

We describe the operation of this model with respect to the route structure in Figure 5. Consider flow between a pair of nodes 1 and 4. Items from 1 first move to their local center S_1. At S_1 we accumulate all items moving from the domain of S_1 (i.e. from points 1,2,3) to the rest of the system (i.e. to points 4 to 11). This accumulation is completed at time = 10 units.

The entire flow from this local center domain to the rest of the system is now moved by rapid transit to the destination local center S_2 (i.e. destination of flow 1-4). If we assume a speed up factor a = 0.2, then this flow reaches S_2 at time = 10 + 0.2 * 23 = 14.6 units. From S_2 this flow is distributed to the destination points and reaches 4 at 14.6 + 6 = 20.6 time units. Thus the flow from 1 to 4 now takes 20.6 time units (as contrasted with 27.6 units for the CSS system).

Figure 6 also lists the travel times between customers in Figure 5. The guaranteed time for the system is the maximum of these times and is 33 units (as against 36 for CSS).

Thus, in this model, travel occurs in three phases. In the first phase,

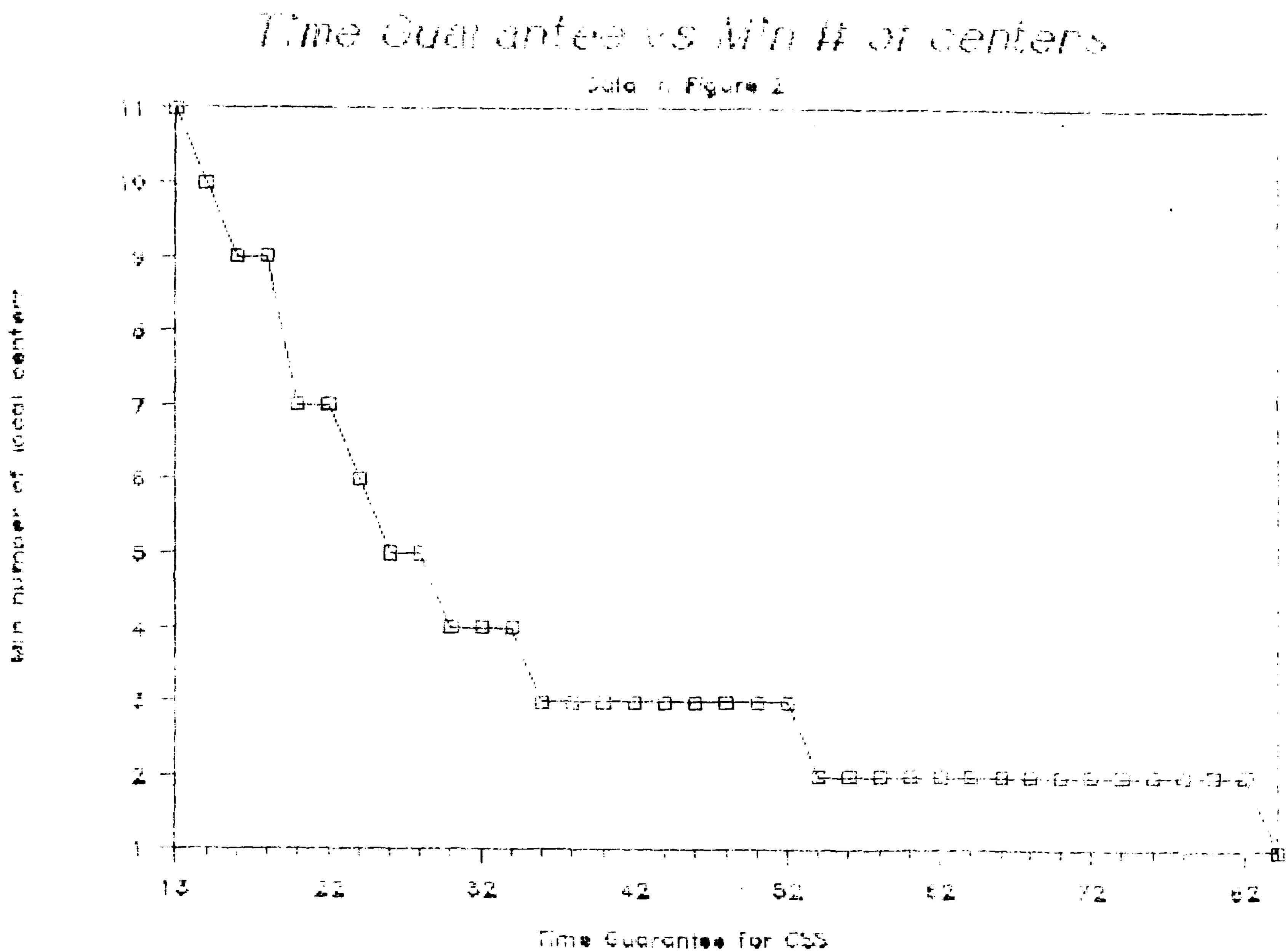


Figure 4: Min number of local centers versus time guarantee for the CSS problem in Figure 2.

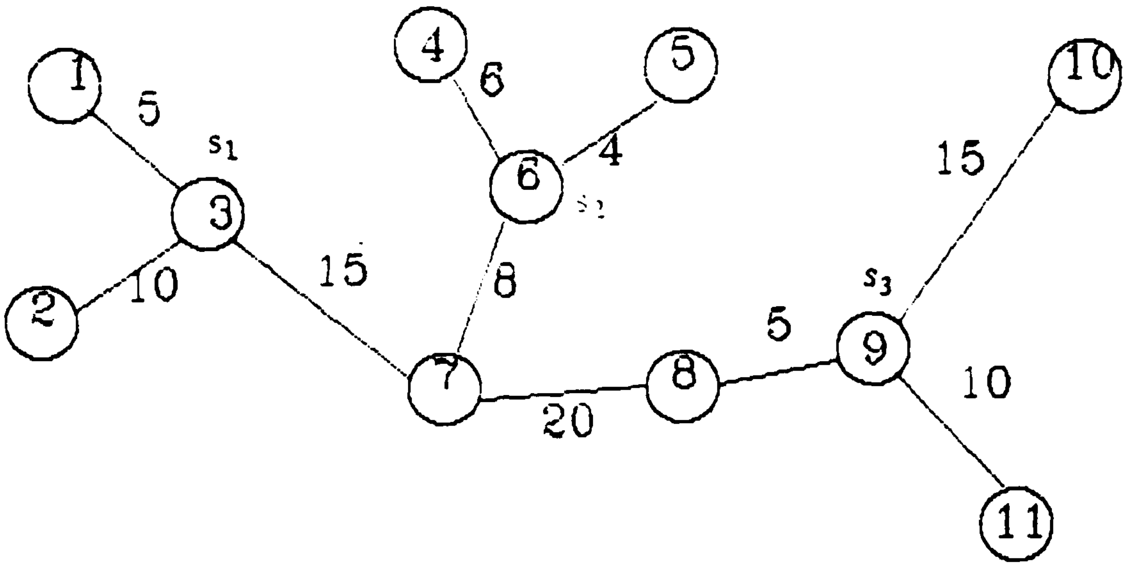


Figure 5: Example DSS with three local centers.

	1	2	3	4	5	6	7	8	9	10	11
1	0	20	10	20.6	18.6	14.6	22.6	23	18	33	28
2		0	10	20.6	18.6	14.6	22.6	23	18	33	28
3			0	20.6	18.6	14.6	22.6	23	18	33	28
4				0	12	8	16	19.6	14.6	29.6	24.6
5					0	8	16	19.6	14.6	29.6	24.6
6						0	16	19.6	14.6	29.6	24.6
7							0	19.6	14.6	29.6	24.6
8								0	15	30	25
9									0	30	25
10										0	25
11											0

Figure 6: DSS travel time between all point pairs using center locations in Figure 5. Time guarantee = 33 units.

flows move from their source to the allocated center. At the local centers, we consolidate flows from a local center i to centers 1,2,3,i-1,i+1,...,k and send these consolidated flows, by rapid transit, to their corresponding centers. This is done independently by each of the local centers. When these flows reach their destination centers, they are sorted and distributed to points in the domain of the center.

The objective function value, for a given location of k centers and allocation of customers to centers = Max $_{\text{all }(i,j)}$ $(d(i,S_i) + a\, d(S_i,S_j) + d(j,S_j))$. Thus the problem is to locate k centers and allocate customers to centers so as to minimize this objective function value.

3.0.1 Resource Requirements

The operation described above implies that we have sorting equipment at each of the local centers. Thus we have k sets of sorting equipment that would enable each of the centers to independently process flows out of points in their domain. This independence also implies that we have k * (k - 1) sets of rapid transit resource units to maintain independence of these consolidated flows.

3.1 Developing the algorithm

In this section we develop optimal algorithms for locating k centers on nodes of a tree graph and allocating remaining nodes to centers such that we

Minimize $(\max\ (i,j)\ (d(i,S_i) + a\, d(S_i,S_j) + d(j,S_j)))$

This problem is solved in four steps

(1) We assume an objective function value T and use it to restrict the possible node allocations to a center.

We provide an algorithm that determines locations of k centers with objective function value <= T. Such an algorithm can be used to solve the optimization problem by a log search over possible values of T (T <- L (i.e. longest path)).

(2) We show that there exists a global center on the tree graph with the property that solving CSS problem with this global center location and objective function <- T/2 yields the solution to the DSS model. This is proved in theorem 1.

(3) We determine possible locations of the global center on the tree and show that it is sufficient to examine certain locations on the tree for a given T. The number of such possible locations for a given T is shown to be polynomial i.e. $O(n^3)$. Algorithm GLOBAL provides these possible locations for the global center.

(4) Finally, we use the CSS algorithm for different values of T and global center locations to solve the problem in this section.

3.2 Algorithm to solve the problem

(1) Consider some setting of the global center. The possible locations that are sufficient to be considered are provided by algorithm GLOBAL.

(2) For the given global center location, solve the CSS problem to yield a k-center location that minimizes the objective function, say R.

This implies a solution to our problem with objective function value <- 2 R.

(3) Examine all global centers and pick that location that yields minimum objective function value. This implies that we repeat step 2 for all of the global center locations in GLOBAL.

The algorithm yields the optimal k-center location that minimizes guaranteed distribution time between centers in a DSS type distribution system. This is because we know that one of the global centers generated by

GLOBAL enables theorem 1 to be used and hence yields the optimal solution to the model with independent flows between centers.

Running time of this algorithm = $O(n^3 * \log(L) * f(n))$

where f(n) is the running time for the covering problem on nodes of the tree, f(n) is $O(n^2)$ using Kolen'82 or Broin and Lowe '86.

3.2.1 Effect of restricting objective function value

We consider the possible locations of k centers that yield an objective function value of <= T. If the objective function value is to be attained, we can restrict the possible values based on the distances between nodes in the tree structure. Since any pair of nodes could determine a possible subtree radius, the possible values of T are obtained by picking subsets of four nodes and using the unique path between the centers to calculate the time required to distribute the flow. Thus there are n^4 possible values for T.

If we set a = 1, objective function value = longest path. Therefore, we generate an upper bound to T. Thus, T <= Longest path.

If we locate a center at each of the nodes, the objective function value = a (Longest path). Thus, we have a lower bound to T, T >= a (Longest path).

Therefore, a (Longest path) <= T <= Longest path.

3.2.2 Existence of a global center

In this section we show that there exists a global center on the tree such that solving the CSS model with this center would yield a solution to the model under consideration.

Theorem 1: There is a location of the global center on the tree such that the CSS model also provides locations of the centers that minimize the DSS objective.

Proof: Consider a location of k centers on the tree graph. Let the maximum value be attained between nodes i and j. Consider a point c on this path which is the same time away from nodes i and j i.e. $d(i,x_i) + a\, d(x_i,c) = d(j,x_j) + a\, d(x_j,c)$.

Solve the CSS problem with this location for the global center and setting the objective function $<= T/2$. We demonstrate that the earlier solution is also optimal to this problem.

Case 1: Clearly points allocated to centers on the tight path are still $<= T/2$ away. So those center locations ensure accumulation at the global center in $<= T/2$.

Case 2: For centers that are not on this tight path, we show that this is still true.

Let x_s be one such center. Also, let c be on the path $P(x_j,x_s)$ and s be a node allocated to x_s. We consider the distance $= d(s,x_s) + a\, d(x_s,c)$. Since it is not tight, we have

$$d(s,x_s) + a\, d(x_s,c) + a\, d(c,x_j) + d(j,x_j) <=$$
$$d(i,x_i) + a\, d(i,c) + a\, d(c,x_j) + d(j,x_j)$$

Hence, $d(s,x_s) + a\, d(x_s,c) <= d(i,x_i) + a\, d(i,c) = T/2$

Figure 7 shows the above argument. Thus we have a solution to the CSS model with this global center for which the solution to the DSS problem is feasible. Furthermore, any k-center location that yields a solution of value R to the CSS problem implies a solution to DSS of value $<= 2$ R. This implies that since T was optimal to the DSS model, $R = T/2$ is optimal to the CSS problem solved for global center c.

Hence the result.

3.2.3 Possible global center locations

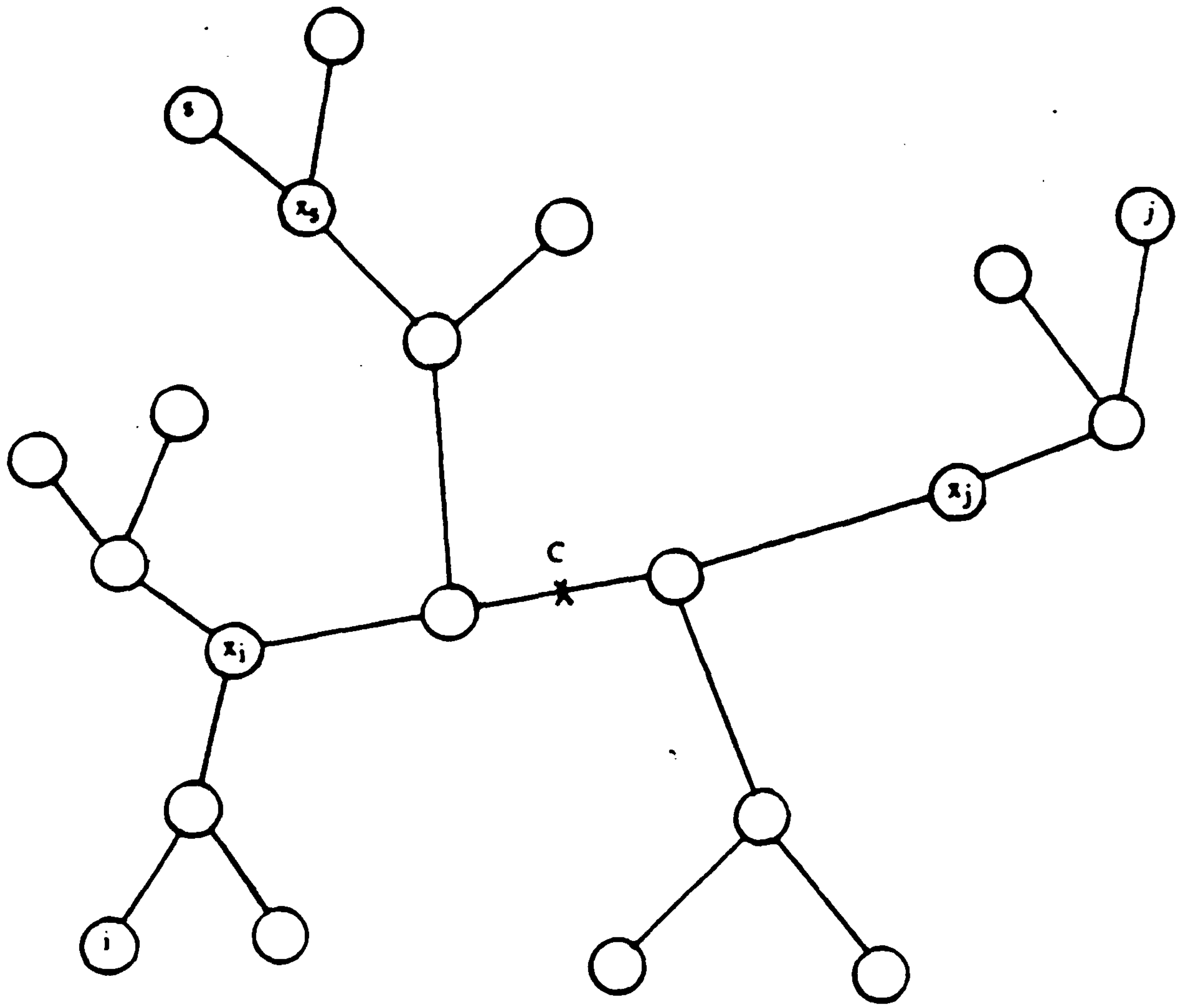


Figure 7: Existence of a global center C optimizing DSS.

The result in the section 3.2.1 implies that we can use the CSS algorithm to solve the DSS problem if we can identify the correct location of the global center. In this section we show that it is sufficient to examine only n^3 possible locations for the global center on the tree for a given T.

Each pair of nodes can potentially form a pair of centers in a tight path of length T. Also, at each of these nodes we could have n-2 possible radius values for its domain. For each radius r_i at node i, there is a potential global center located at a distance of $(T/2 - r_i)/a$ along the edge between the node pair. Thus we have 2 * (n - 2) possible global center settings for a given node pair of potential centers.

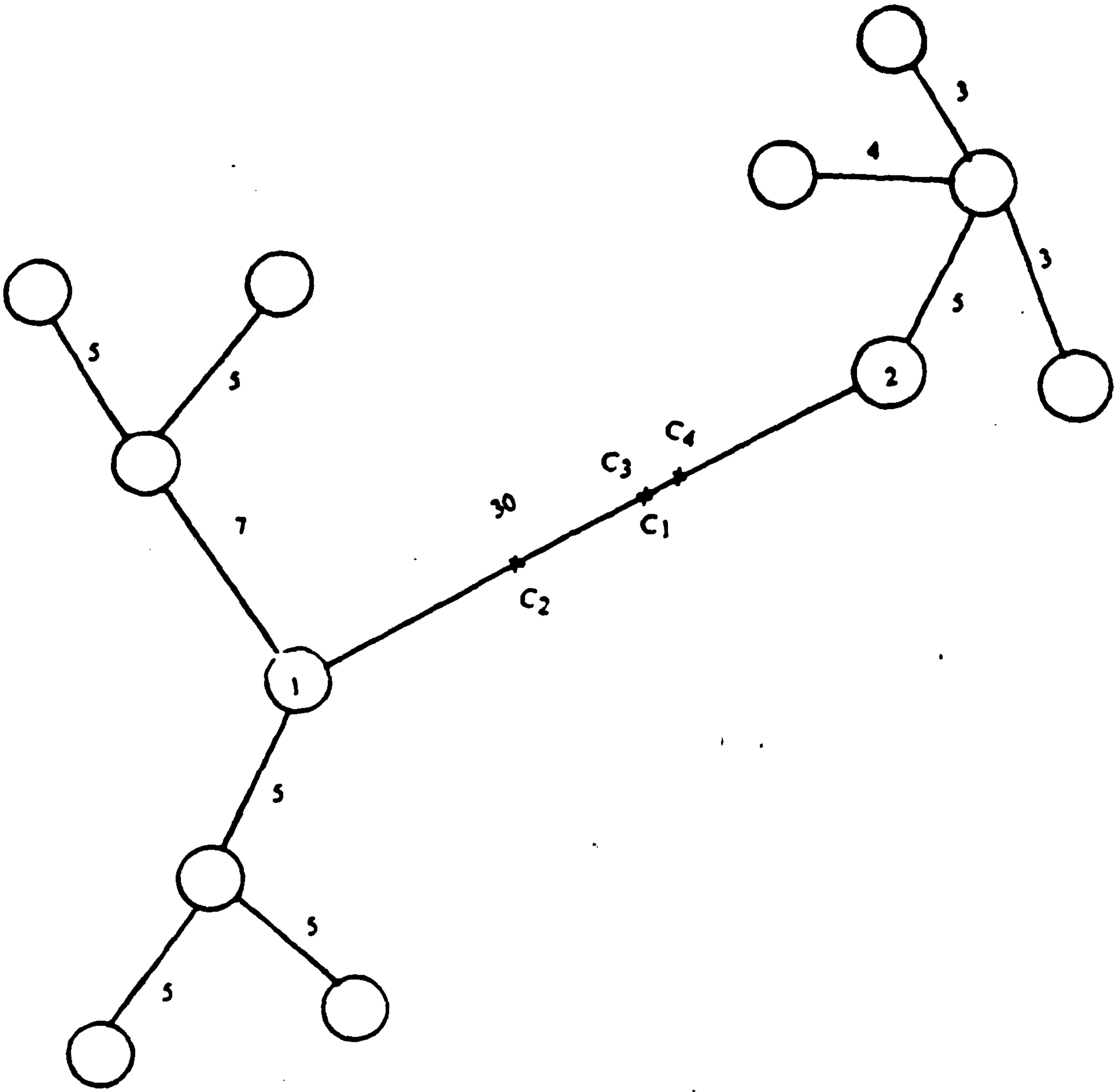
Figure 8 shows a pair of potential centers (1,2) and potential global center locations for T = 30. At node 1, we have possible radii $r_1 = 7$, $r_2 = 10$ for which we have corresponding global centers at $C_1 = (15 - 7)/ 0.5 = 16$, $C_2 = (15 - 10)/0.5 = 10$ units from node 1 along (1,2). Node 2 has possible radii settings $r_3 = 8$, $r_4 = 9$ with global centers at $C_3 = (15 - 8)/0.5 = 14$ and $C_4 = (15 - 9)/0.5 = 12$ units from node 2 along (1,2). For each of the radius settings we have the four potential global center locations along (i,j) as indicated.

However, there are (n * n) possible node pairs. Across all possible node pairs we have $(n * n - 1) * n = O(n^3)$ possible locations for the global center. Thus, the possible locations of the global center are as follows

Algorithm GLOBAL

(1) For each pair of nodes (i,j), step 2 provides the 2 n possible center locations on the path joining nodes i and j (unique path due to tree structure).

(2) For each node k allocated to node j, the corresponding center location is



T = 30

$C_1 = (15 - 7)/ 0.5 = 16$ units from 1

$C_2 = (15 - 10)/ 0.5 = 10$ units from 1

$C_3 = (15 - 8)/ 0.5 = 14$ units from 2

$C_4 = (15 - 9)/ 0.5 = 12$ units from 2

Figure 8: Potential global center locations on (1,2).

a distance $(T/2 - d(k,j))/a$ from node j, along the path linking j and i. Thus, there are n such possible center locations.

For each node k allocated to node i, we have a similar set of n locations on the path linking i and j.

Thus, we have $O(n * n * 2n) = O(n^3)$ possible global center locations

3.2.4 Effect of restricting local centers locations

If the local center locations are restricted to be at g(n) nodes (g(n) > k, else trivial), then we can use this information to reduce running time of the algorithm in GLOBAL. Possible locations to be examined for the global center, 2n locations for each pair of nodes in g(n), is now $O(n (g(n))^2)$.

Thus the running time of the algorithm with local centers restricted to be on g(n) nodes is $O(n (g(n))^2 \log(L) f(n))$.

3.2.5 Extension to k centers not restricted to nodes

If the k centers are not restricted to nodes of the tree, then it can be verified that for each of the subtrees formed, the center would be located at the center of the longest path in the subtree. Therefore, the potential locations of centers are provided by the half way points on all paths between node pairs of the tree. Thus, there are $O(n^2)$ potential locations for the k centers on the tree.

For each pair of potential centers, we can determine the possible locations for the global center given an objective function value of T as in GLOBAL. Therefore the algorithm would proceed as in DSS. The only difference in the A matrix created would be that the center locations that are not nodes would not require covering by any of the k-centers hence would not have covering constraints associated with them.

4.0 **Conclusions and Extensions**

We provide algorithms for locating k centers on a tree graph so as to have a guaranteed delivery time (T) between any two nodes in the graph. We also provide two different models with different distribution strategies between points in the graph.

Several extensions could be envisaged for this work. Some possible extensions are as follows

(1) Different flow classes with differing levels of priority.

(2) Different travel times between centers depending on the kinds of resources allocated. The location of centers could thus be combined with a resource allocation problem to determine the best allocation of resources (airplanes) so that the guaranteed time (T) is minimized.

(3) Performance of heuristics for solving the guaranteed time location problem on general graphs which use the algorithms developed for tree graphs on some spanning tree.

BIBLIOGRAPHY

(1) Barany, I., Edmonds, J. and Wolsey, L., "Packing and Covering a tree by subtrees", CORE Discussion Paper No. 8434, CORE, Belgium, July 1984

(2) Broin, M.W. and Lowe, T.J., "A Dynamic Programming Algorithm for Covering Problems with (Greedy) Totally Balanced Constraint Matrices", SIAM Journal of Algebriac and Discrete Methods, Vol. 7, No. 3, pg 348-357, July 1986

(3) Choi, H.A. and Hakimi, S.L., " Scheduling File Transfers for Trees and Odd Cycles", SIAM Journal of Computing, Vol. 16, No. 1, pg. 162-168, Feb 1987

(4) Coffman E.G., Garey M.R., Johnson D.S. and LaPaugh, A.S., " Scheduling File Transfers", SIAM Journal of Computing, Vol. 14, No. 3, pg 744-780, August 1985

(5) Goldman, A.J., "Optimal Location for Centers in a Network",

Transportation Science, Vol. 3, pg 352-360, 1969

(6) Hakimi, S.L. and Maheshwari, S.N., "Optimal Locations for Centers in Networks", Operations Research, Vol. 20, pg 967-973, 1972

(7) Iyer, A.V., Ratliff, H.D. and Vijayan, G., "Modeling and Analysis of Parallelism in Assembly Operations", (in Preparation), 1987

(8) Kariv, O. and Hakimi, S.L., "An Algorithmic Approach to Network Location Problems. I: The p-Centers", SIAM Journal of Applied Mathematics, Vol. 37, No. 3, pg 513-538, December 1979

(9) Kolen, A.W.J, "Solving Covering Problems and the Capacitated Plant Location Problem on Trees", Working Paper, Mathematisch Centrum, Amsterdam, 1982

(10) O'Kelly, M. E., "The Location of Interacting Hub Facilities", Transportation Science, Vol. 20, No. 2, pg. 92-106, May 1986

(11) Tansel, B.C., Francis, R.L. and Lowe, T.J., " Location on Networks: A Survey,Parts I and II: The p-Center and p-Median Problems", Management Science, V. 29, No. 4,pg 498-511, April '83